

5.6 Adaptive Clocking System for Improved Power Efficiency in a 28nm x86-64 Microprocessor

Aaron Grenat¹, Sanjay Pant¹, Ravinder Rachala¹, Samuel Naffziger²

¹AMD, Austin, TX, ²AMD, Fort Collins, CO

In high-performance microprocessor cores, the on-die supply voltage seen by the transistors is non-ideal and exhibits significant fluctuations. These supply fluctuations are caused by sudden changes in the current consumed by the microprocessor in response to variations in workloads. This non-ideal supply can cause performance degradation or functional failures. Therefore, a significant amount of margin (10-15%) needs to be added to the ideal voltage (if there were no AC voltage variations) to ensure that the processor always executes correctly at the committed voltage-frequency points. This excess voltage wastes power proportional to the square of the voltage increase.

Several techniques have been proposed and implemented either to mitigate or compensate for supply noise. The techniques include the following: (1) Improving the supply network impedance by addition of decoupling capacitance or better packaging. (2) Static voltage margining: set the VRM output to be higher by some amount (guard band); this method costs significant additional power. (3) VR load-lining: the VRM supply is increased during periods of low processor activity to provide voltage headroom for potential supply droops. This approach can recover in the range of one-quarter to one-third of the F_{max} margin with the downside of increased power during low-load conditions. This method may be susceptible to workloads with both high average power and large transients [7]. (4) Architectural techniques for increasing the ramp-up or ramp-down time of current surges/drops of the processor by throttling the instructions issued [3, 5, 6]. (5) Adaptive clocking: clock period adjustment in response to supply variation [1, 2]; Intel's PLL clock adjustment in Nehalem appears to fall into this category of supply-droop compensation. These methods adaptively tune the processor clock cycle based on the supply droop. In Intel's solution, the supply voltage of the processor core is mixed into the supply of the PLL VCO, thus increasing the clock period during supply droops and decreasing the clock period during supply overshoots. These five approaches address only part of the droop spectrum, are limited in their efficacy by recovering only a small part of the frequency loss (5), or have significant costs such as packaging costs (1), IPC loss (4) or power increases (2 and 3).

The adaptive clocking technique implemented in this AMD x86-64 microprocessor core addresses both 1st and 2nd droops and can recover a larger percentage of the frequency loss at virtually no cost. The design consists of two circuits (Fig. 5.6.1). A configurable droop detector (e.g., 2.5% or 5% of supply voltage) which detects that the supply is drooping, and a digital frequency synthesizer which increases the clock period by a configurable amount (e.g., 5% or 10%). When a voltage droop is detected, the clock runs at a lower frequency (generated by the DFS) until the supply voltage comes back above the threshold (Fig. 5.6.2). This avoids timing failures due to the decreased voltage, which translates to higher overall frequency. The interface between the two circuits is asynchronous because the droop-threshold crossing can occur at any time relative to the clock edge. Fig. 5.6.7 shows the placement on die.

The droop detector circuit uses a DLL to compare the locked output of a delay line with the phase of the PLL output clock. It has 3 configuration bits available for programming a "droop threshold" below which "stretch" events are triggered, which is used to optimize circuit performance post silicon. To detect a drop in the supply, a programmable phase (determined by the droop threshold setting) of the DLL is compared to phase-0 using a simple phase detector. Whenever the core supply is below the set droop threshold, a stretch signal is asserted that requests the clock-phase picker to pick appropriate phases in the phase generator to generate a stretched clock. This droop-detected signal is asynchronous; it is synchronized using a high- t synchronizer with a configurable number of latches into the picker clock domain to minimize latency in the reaction time to generate a slowed clock. The reaction time to begin stretching the clock is one to three cycles depending on the configuration.

To create an instantly reduced clock frequency, a phase generator with a DLL is used to generate 20 phases of the PLL output clock. The DLL operates at a regulated voltage. To support finer stretch amounts, the 20 clock phases are converted to 40 phases using an interpolator chain. In the event of a droop, a glitch-less clock picker circuit takes the 40 DLL phases and generates a stretched clock by selecting appropriate phases. The clock picker always performs a complete loop through all the phases before selecting the 0th phase going back to the default clock frequency. The additional logic in the clock path does lead to additional jitter measured in silicon as 0.5-1% of cycle time; however, this jitter is encountered only when the clock is stretched and the core already is operating at a lower frequency. Both the phase-generator and droop-detector DLLs were verified to be stable under intentional dynamic voltage-frequency transitions as well as power-supply noise transients and input-clock jitter.

An on-die power supply monitor (PSM) circuit is used to monitor the voltage rail during the stretch event. The PSM is a high-speed time-to-digital converter that counts the number of gate transitions in a clock period [4]. Once calibrated, the PSM result can be translated directly to voltage at the transistor. Fig. 5.6.3 shows voltage at the transistor plotted against time, indicating stretch being triggered as the voltage is falling in a droop event.

The adaptive clocking system provides a benefit of up to 9% V_{min} reduction. Fig. 5.6.4 shows the power benefit across the operating range with the adaptive clocking system enabled when running production speed-binning workloads. At lower voltages and frequencies, the amount of di/dt-induced high-speed transients is less than at higher voltage. This data is taken on a high-power desktop platform. Increases in series inductance in lower-power platforms not designed for such high currents may increase the adaptive clocking system's performance at lower voltage and frequency.

The optimal setting for droop threshold presents a balance between stretching the clock aggressively for even small droop events (thereby reducing the average frequency) and using a higher threshold so stretch events are rare (average frequency is not affected) but less margin is recovered. For typical applications, we determined a droop threshold value of 2.5% to be most beneficial to overall core performance for most workloads. Fig. 5.6.5 displays the amount of stretching incurred with different droop thresholds across workloads. At 1.25%, many workloads spent longer periods in the slowed-clock state, leading to additional performance loss. It also shows average core frequency while running standard application suites. Fig. 5.6.6 shows the system benefit (reduced V_{min}) with programmable "droop threshold" and "stretch amount" parameters.

In summary, this adaptive clocking system enables the reduction of voltage at a given frequency by 3% to 6%, resulting in core power efficiency increasing by 7% to 15% (Fig. 5.6.4).

Acknowledgments:

Visvesh Sathe, Stephen Kosonocky, and Frank Huang

References:

- [1] T. Fischer, *et al.*, "A 90nm Variable-Frequency Clock System for a Power-Managed Itanium®-Family Processor," *ISSCC Dig. Tech. Papers*, pp. 294-295, 2005.
- [2] N. Kurd, *et al.*, "Next Generation Intel Core Micro-Architecture (Nehalem) Clocking," *IEEE J. Solid-State Circuits*, vol. 44, no. 4, pp. 1121-1129, 2009.
- [3] M. Floyd *et al.*, "Introducing the Adaptive Energy Management Features of the Power7 Chip," *IEEE Micro*, vol. 31, no. 2, pp.60-75, March-April 2011
- [4] K. Gillespie, *et al.*, "Steamroller, an x86-64 Core Implemented in 28nm Bulk CMOS," to appear in *ISSCC Dig. Tech. Papers*, 2014.
- [5] M. Gupta, *et al.*, "An Event-Guided Approach to Reducing Voltage Noise in Processors," *IEEE Design, Automation and Test Conf.*, pp. 160-165, 2009.
- [6] V.J. Reddi, *et al.*, "Predicting Voltage Droops Using Recurring Program and Microarchitectural Event Activity," *IEEE Micro*, vol. 30, no.1, pp.110, 2010
- [7] X. Zhang, *et al.*, "A Novel VRM Control with Direct Load Current Feedback," *IEEE Applied Power Electronics Conf.*, pp. 267-271, 2004.
- [8] J. Tschanz, *et al.*, "Adaptive Frequency and Biasing Techniques for Tolerance to Dynamic Temperature-Voltage Variations and Aging," *ISSCC Dig. Tech. Papers*, pp. 292-293, 2007.

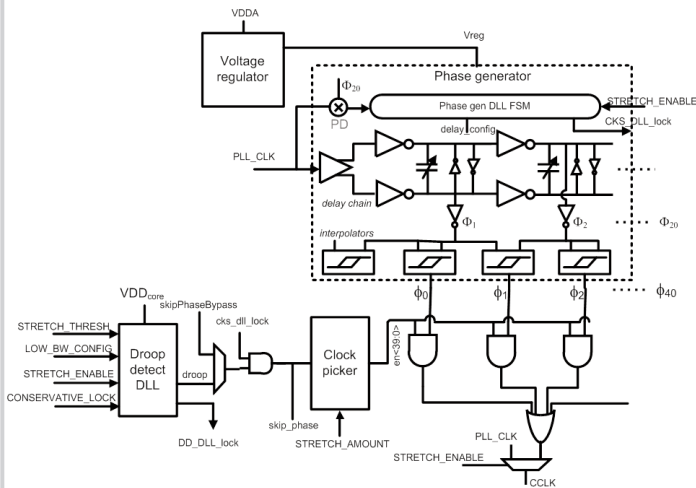


Figure 5.6.1: Block diagram of the adaptive clocking system.

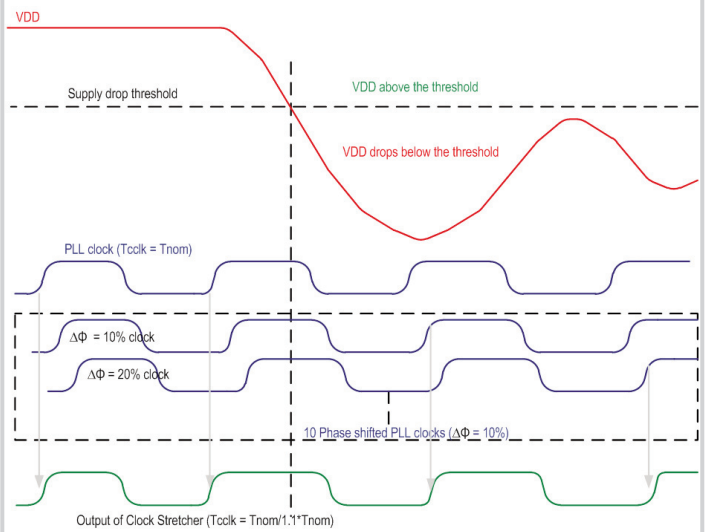


Figure 5.6.2: Principle of adaptive clocking system operation.

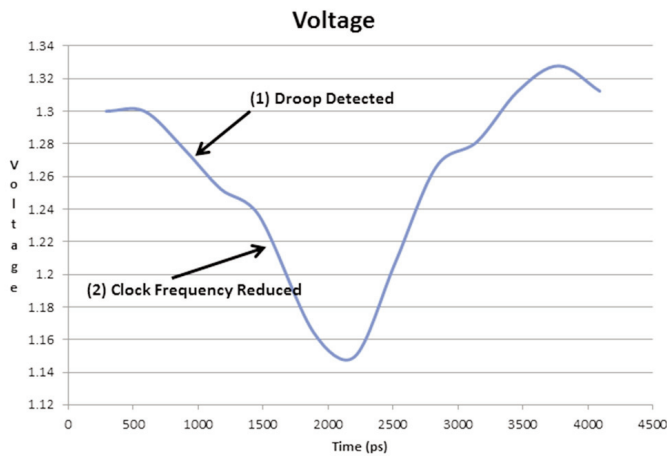
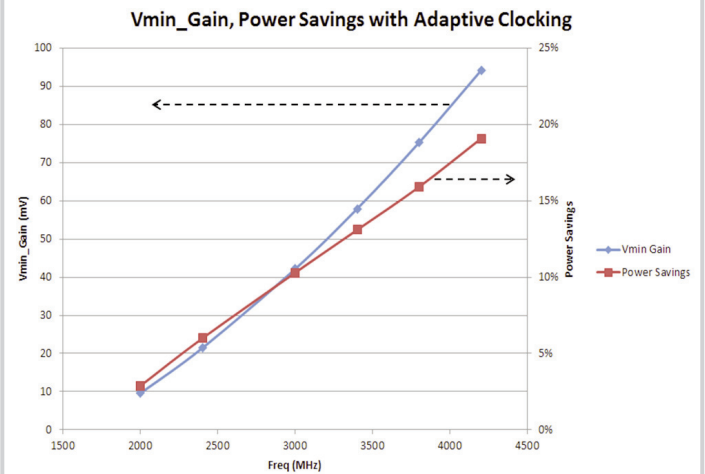


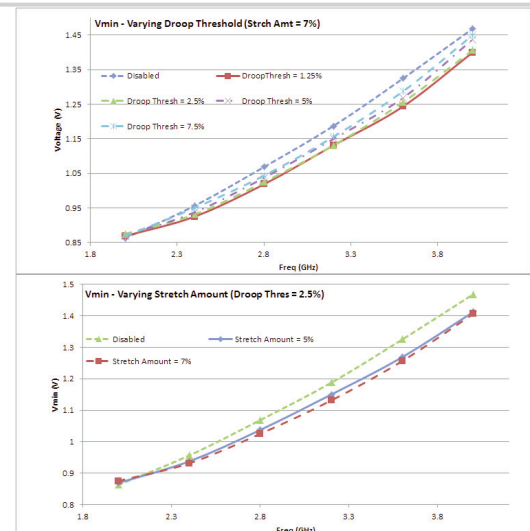
Figure 5.6.3: PSM measurements of the voltage at the transistor.

Figure 5.6.4: V_{min} and corresponding power savings with 7% stretch and 2.5% droop threshold.

DroopThreshold	1.25%	2.50%	3.75%	5.00%
Stretch%	0.50%	0.22%	0.02%	0.0002430%

	CU0-avg-freq	CU1-avg-freq
3DMark	3398	3396
CineBench	3397	3394
iTunes_aac	3396	3393
iTunes_mp3	3396	3394
POVRay	3394	3391
WinRAR	3400	3400

Figure 5.6.5: (Top) Number of cycles stretched while running standard workloads; (Bottom) Average frequency observed on standard apps with adaptive clocking enabled (input frequency = 3400 MHz).

Figure 5.6.6: V_{min} benefit with varying "droop threshold" and "stretch amount" settings.

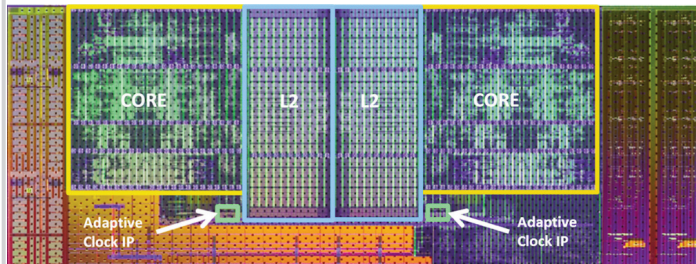


Figure 5.6.7: Die plot showing adaptive clock IP placement and corresponding cores.